J Glob Optim (2007) 37:557–569
DOI 10.1007/s10898-006-9063-7

ORIGINAL PAPER

A robust algorithm for quadratic optimization under quadratic constraints

Hoang Tuy · N. T. Hoai-Phuong

Received: 21 June 2006 / Accepted: 23 June 2006 / Published online: 9 August 2006

Abstract Most existing methods of quadratically constrained quadratic optimization actually solve a refined linear or convex relaxation of the original problem. It turned out, however, that such an approach may sometimes provide an infeasible solution which cannot be accepted as an approximate optimal solution in any reasonable sense. To overcome these limitations a new approach is proposed that guarantees a more appropriate approximate optimal solution which is also stable under small perturbations of the constraints.

Keywords Nonconvex global optimization · Quadratic optimization under quadratic constraints · Branch-reduce-and-bound successive incumbent transcending algorithm · Essential optimal solution · Robust solution

1 Introduction

A large number of problems arising from applications in diverse fields can be formulated as nonconvex quadratically constrained quadratic optimization problems of the form

$$(\text{QQP}) \qquad \begin{array}{ll} \min & f_0(x) \\ \text{s.t.} & f_k(x) \geq 0 \; k = 1, \ldots, m, \quad x \in C, \end{array}$$

where $C \subset \mathbb{R}^n$ is a polyhedron, and each $f_k(x)$, $k = 0, 1, \ldots, m$, is a quadratic function:

$$f_k(x) = \sum_{i<j} c_{ij}^k x_i x_j + \sum_i c_i^k x_i^2 + \sum_i d_i^k x_i + b_k.$$

We will assume that C is contained in a box $[a, b] := \{a \leq x \leq b\} \subset \mathbb{R}_+^n$.

H. Tuy (✉) · N. T. Hoai-Phuong
Institute of Mathematics, 18 Hoang Quoc Viet Road, Hanoi, Vietnam
e-mail: htuy@math.ac.vn

N. T. Hoai-Phuong
e-mail: htphuong@math.ac.vn

As is well known, linear mixed 0-1, bilinear, fractional, polynomial, bilevel, generalized linear complementarity problems, can be reformulated as special cases of (QQP) [6, 13]. In view of its interest both from a theoretical and practical viewpoint, this problem has in recent years attracted the attention of many researchers. Among the best known numerical studies devoted to (QQP), we should mention the works [1, 2, 4, 7, 8–12].

In all these works, however, robustness is never a matter of serious concern. Most algorithms so far developed in the literature solve a more or less refined relaxation of (QQP) and provide an approximate optimal solution which is believed, without any guarantee, to be close to the exact optimum. However, as has been shown in [16, 17] (see Section 2 below) such an approximate optimal solution may turn out in certain cases to be so far from the exact optimum that it can hardly be accepted as an approximation of the latter. This poses the necessity to re-examine the approximation concept so far commonly used and stresses the importance of robustness for practical implementation of global optimization methods. Motivated by these considerations, a robust approach to nonconvex global optimization has been proposed in [16] and further improved in [17].

In the present paper, we will specialize this robust approach to (QQP) and demonstrate its practical applicability on nontrivial examples taken from the literature. Basically, the approach consists in transforming (QQP) into a monotonic optimization problem and applying a special procedure of monotonic optimization earlier developed in [14, 15] to compute a robust optimal solution to a problem derived from the original problem just by omitting the isolated, hence instable, feasible solutions.

The paper is organized as follows. After the Introduction, in Sect. 2 we show a potential drawback of the approximation scheme underlying most existing methods of nonconvex quadratic optimization. Section 3 deals with the conversion of (QQP) into a form amenable to monotnic optimization. Section 4 introduces the new concept of essential optimality which should be more easily implementable and more appropriate than the usual concept of optimality. The search for an essential optimal solution is achieved by successive application of a special procedure for transcending an incumbent. Section 5 describes this procedure. In Section 6 this procedure is incorporated into an algorithm for solving (QQP) to be referred to as the Successive Incumbent Transcending (SIT) algorithm. Section 7 closes the paper with an instructive numerical example and some preliminary computational results illustrating the practicality of the approach.

2 Drawbacks of common approaches

Since the constraint set of (QQP) is nonconvex, finding a feasible solution is almost as difficult a task as solving the problem itself. Therefore, except in rare special cases, in finitely many steps, a numerical method can only guarantee an approximate optimal solution, i.e. a solution which is close, in some acceptable sense, to the global optimum.

In most approaches so far commonly adopted, the nonconvex feasible set is relaxed to a polyhedron (linear relaxation) or to a convex, easily computable, set (convex relaxation). Given a tolerance $\varepsilon > 0$, any point x that is feasible to the problem

$$\text{(QQP}(\varepsilon)) \qquad \min\{f_0(x) \mid f_k(x) + \varepsilon \geq 0,\ k = 1, \ldots, m,\ x \in C\}$$

is called an ε-*approximate feasible solution*. Then the approximate problem to substitute for (QQP) is to find an ε-*approximate optimal solution*, i.e. an optimal solution of the ε-approximate problem (QQP(ε)). Although this has become a common practice in nonconvex global optimization, most people do not seem to be aware of the pitfall behind this approximation scheme.

Consider for example the quadratic program depicted in Fig. 1, where the objective function is $f(x)$ and the constraints are $g_i(x) \geq 0$ $(i = 1, 2, 3)$, $a \leq x \leq b$. The optimal solution is x^*, while the point $\bar{x}$ is infeasible, but almost feasible:

$$g_1(\bar{x}) = 0, \qquad g_2(\bar{x}) = 0, \qquad g_3(\bar{x}) = -\delta$$

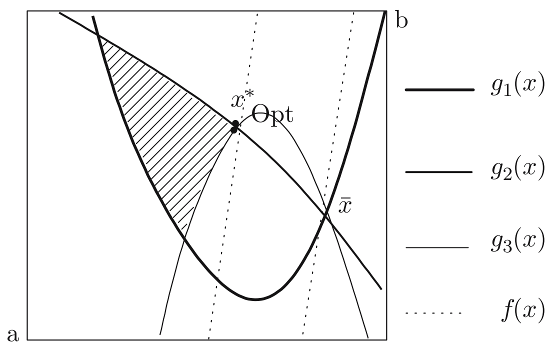
for some small $\delta > 0$. If $\varepsilon > \delta$ then $\bar{x}$ is feasible to the ε-relaxed problem, and will be accepted as an ε-approximate optimal solution, though it is quite far from the exact optimal solution x^*. But if $\varepsilon < \delta$ then $\bar{x}$ is no longer feasible to the ε-relaxed problem and the ε-approximate solution will come close to x^*.

Thus, even for regular problems (problems whose feasible set is the closure of its interior), the ε-relaxation approach may give an incorrect optimal solution if ε is not sufficiently small. The trouble is that in practice we often do not know what exactly means "sufficiently small," i.e. we do not know how small the tolerance should be to guarantee a correct approximate optimal solution.

Furthermore, in many cases an ε-approximate optimal solution, i.e. an optimal solution of QQP(ε), cannot be computed in finitely many steps. Therefore, a second level of approximation is needed. Given $\eta > 0$, a feasible solution $\tilde{x}$ to QQP(ε) such that $f_0(\tilde{x}) \leq \min(\text{QQP}(\varepsilon) + \eta$ is referred to as an (ε, η)-*optimal solution* of (QQP). In finitely many steps only such an (ε, η)-optimal solution can be guaranteed. Again it is often not easy to determine the appropriate values of ε and η to guarantee a correct approximate optimal solution.

Last but not least, most algorithms so far developed for (QQP) proceed by branch and bound, with lower bounding based on outer approximation. It follows from the above discussion that it may quite happen that a partition set chosen for further partitioning in some iteration contains no feasible solution. This may occur, for example, if a partition set M contains no feasible point but an almost feasible point belongng to the approximate feasible set: the lower bound over M is then a finite number and not $+\infty$ as it should be to prevent it from being a candidate for further partitioning. The branch and bound algorithm in such cases may converge to an approximate optimal solution which is infeasible.

Fig. 1 Inadequate ε-approximate optimal solution



3 Monotonic reformulation

A function $f\colon\ \mathbb{R}^n \to \mathbb{R}$ is said to be *increasing* on the orthant $\mathbb{R}^n_+$ if $f(x') \le f(x)$ whenever $0 \le x' \le x$, i.e. $0 \le x'_i \le x_i \quad \forall i = 1, \dots, n$. It is said to be a *d.m. function* on $\mathbb{R}^n_+$ if $f(x) = f_1(x) - f_2(x)$, where f_1, f_2 are increasing on $\mathbb{R}^n_+$.

Clearly a quadratic polynomial of n variables with positive coefficients is increasing on $\mathbb{R}^n_+$. Since every quadratic polynomial can be written as a difference of two quadratic polynomials with positive coefficients, it follows that each function $f_k(x),\ k = 0, 1, \dots,$ is a d.m. function on $\mathbb{R}^n_+$:

$$f_k(x) = f_k^+(x) - f_k^-(x).$$

Therefore, the general problem (QQP) can be rewritten as the d.m. optimization problem

$$\begin{array}{ll} \min & f_0^+(x) - f_0^-(x), \\ \text{s.t.} & f_k^+(x) - f_k^-(x) \ge 0, \quad k = 1, \dots, m,\ x \in C. \end{array}$$

Noting that $f_0^-(a) \le f_0^-(x) \le f_0^-(b)\ \forall x \in [a, b]$, we can also write the problem as

$$\begin{array}{lll} \min & f_0^+(x) + t, & \\ \text{s.t.} & t + f_0^-(x) \ge 0, & \\ & f_k^+(x) - f_k^-(x) \ge 0, & k = 1, \dots, m \\ & x \in C, \quad -f_0^-(b) \le t \le -f_0^-(a). & \end{array}$$

Finally, by changing the notation, one can thus convert (QQP) into the form

$$(P), \qquad \min\{f(x) |\ g(x) \ge 0,\ x \in [a, b]\},$$

where $f(x)$ is an increasing quadratic function:

$$f(x) = \sum_{i<j} c_{ij} x_i x_j + \sum_i c_i x_i^2 + \sum_i d_i x_i \tag{1}$$

with all coefficients being positive, while

$$g(x) = \min_{k=1,\dots,m} \{u_k(x) - v_k(x)\} \tag{2}$$

with $u_k(x),\ v_k(x)$ being increasing quadratic functions such that

$$g_k(x) := u_k(x) - v_k(x) = \sum_{i<j} c^k_{ij} x_i x_j + \sum_i c^k_i x_i^2 + \sum_i d^k_i x_i + b_k. \tag{3}$$

Note that the quadratic functions (1) and (3), are not the same as the original functions $f_k(x)$ in (QQP), although the same notation has been used.

4 Essential optimal solution

From now on we assume that the original problem (QQP) has been converted to the form (P), with $f(x)$ quadratic increasing and $g(x)$ defined as in (1), (2), and (3).

As was argued in Sect. 2, an algorithm giving only an ε-approximate optimal solution may not be quite correct. Furthermore, an isolated optimal solution even if computable is often difficult to implement practically because of its instability under small perturbations of the constraints. Therefore, from a practical point of view only nonisolated feasible solutions should be of interest. This motivates the following definitions.

A nonisolated feasible solution x^* of (P) is called an *essential optimal solution* if $f(x^*) \le f(x)$ for all nonisolated feasible solutions x of (P), i.e. if

$$f(x^*) = \min\{f(x) |\ x \in S^*\},$$

where S^* denotes the set of all nonisolated feasible solutions of (P). Assume

$$\{x \in [a,b] |\ g(x) > 0\} \neq \emptyset. \tag{4}$$

For $\varepsilon \ge 0$, an $x \in [a,b]$ satisfying $g(x) \ge \varepsilon$ is then called an *ε-essential feasible solution*, and a nonisolated feasible solution $\bar{x}$ of (Q) is called an *essential ε-optimal solution* if it satisfies

$$f(\bar{x}) - \varepsilon \le \inf\{f(x) |\ g(x) \ge \varepsilon,\ x \in [a,b]\}. \tag{5}$$

Clearly for $\varepsilon = 0$ a nonisolated feasible solution which is essentially ε-optimal is optimal.

The search for an essential ε-optimal solution of (P) can be carried out according to the following SIT *scheme*: starting from an initial essential feasible solution (the incumbent, i.e. the best so far known), find a better essential feasible solution (transcend the incumbent), then reiterate the operation until an evidence is obtained that no better feasible solution than the current best exists.

Let γ be the objective function value of an essential feasible solution (of course $\gamma \le f(b)$). Given $\varepsilon > 0$ we want to find, if possible, an essential feasible solution x with $f(x) \le \gamma - \varepsilon$.

If $f(a) \ge \gamma - \varepsilon$, then, since f is increasing, $f(x) \ge \gamma - \varepsilon\ \forall x \in [a,b]$, so there is no $x \in [a,b]$ with an objective function value less than $\gamma - \varepsilon$.

If $f(a) < \gamma - \varepsilon$ and $g(a) > 0$, then a is an essential feasible solution with objective function value less than $\gamma - \varepsilon$.

Therefore, barring these two cases, we can assume that

$$f(a) < \gamma - \varepsilon, \qquad g(a) \le 0. \tag{6}$$

Consider then the following auxiliary problem associated with γ :

$$(\mathrm{Q}/\gamma) \qquad\qquad \max\{g(x) |\ f(x) \le \gamma - \varepsilon,\ x \in [a,b]\}.$$

For our purpose of robust optimization an important feature of (Q/γ) is that, due to the assumption that $f(x)$ is continuous and increasing, this problem has no isolated feasible point. Solving (Q/γ) is therefore simpler than solving the original problem (P).

Proposition 1 *Assume* (6).

(1) *Any feasible solution x^0 of (Q/γ) such that $g(x^0) > 0$ is a nonisolated feasible solution of* (P) *with $f(x^0) \le \gamma - \varepsilon$. In particular, if* max (Q/γ) > 0 *then the optimal solution $\hat{x}$ of* (Q/γ) *is a nonisolated feasible solution of* (P) *with $f(\hat{x}) \le \gamma - \varepsilon$.*
(2) *If* max (Q/γ) $< \varepsilon$ *and $\gamma = f(\bar{x})$ for some nonisolated feasible solution $\bar{x}$ of* (P), *then $\bar{x}$ is an essential ε-optimal solution of* (P). *If* max (Q/γ) $< \varepsilon$ *and $\gamma = f(b) + \varepsilon$ then the problem* (P) *is ε-essentially infeasible* (i.e. has no essential feasible solution).

Proof (1) Since $g(a) \le 0 < g(x^0)$, we have $x^0 \neq a$ and every $x = a + \lambda(x^0 - a)$ with $0 \le \lambda \le 1$ satisfies $a \le x \le x^0$. Then for every λ sufficiently close to 1, i.e. every x

sufficiently close to x^0, we have $g(x) > 0$, so x is a feasible solution of (P). Hence, x^0 is a nonisolated feasible solution of (P). Furthermore, $f(x^0) \leq \gamma - \varepsilon$ because x^0 is feasible to (Q/γ).

(2) If $\max(\mathrm{Q}/\gamma) < \varepsilon$ then

$$\varepsilon > \sup\{g(x)|\, f(x) \leq \gamma - \varepsilon,\ x \in [a,b]\}$$

so every $x \in [a,b]$ such that $g(x) \geq \varepsilon$, must satisfy $f(x) > \gamma - \varepsilon$. Therefore, if $\gamma = f(\bar{x})$ then

$$\inf\{f(x)|\, g(x) \geq \varepsilon,\ x \in [a,b]\} \geq f(\bar{x}) - \varepsilon,$$

i.e. $\bar{x}$ is an essential ε-optimal solution. If $\gamma = f(b) + \varepsilon$, then $\{x \in [a,b]|\, g(x) \geq \varepsilon\} = \emptyset$, i.e. the problem is ε-essentially infeasible. □

Thus, solving (Q/γ) gives information about whether or not an essential feasible solution x exists such that $f(x) \leq \gamma - \varepsilon$.

5 Solving (Q/γ)

The procedure we propose for solving (Q/γ) is a Branch-Reduce-and-Bound (BRB) algorithm involving three basic operations: branching, reducing (the partition sets) and bounding as follows:

(1) Branching proceeds by successive rectangular partition of the initial box $M_0 = [a,b]$ according to an *exhaustive* subdivision rule, i.e. such that any infinite nested sequence of partition sets generated through the algorithm shrinks to a singleton. A popular exhaustive subdivision rule is the *standard bisection*.

(2) Reducing consists in reducing the size of a partition set $M = [p,q] \subset [a,b]$ without losing any feasible solution currently still of interest. The box $[p',q']$ obtained that way from M is referred to as a *valid reduction* of M.

(3) Bounding consists in estimating an upper bound $\beta(M)$ for $g(x)$ over the feasible portion of (Q/γ) contained in the valid reduction $[p',q']$ of a given partition set $M = [p,q]$.

5.1 Valid reduction

At a given stage of the BRB algorithm for (Q/γ), let $[p,q] \subset [a,b]$ be a box generated during the partitioning procedure and still of interest. The search for a nonisolated feasible solution of (Q) in $[p,q]$ such that $f(x) \leq \gamma - \varepsilon$ can then be restricted to the set $B_\gamma \cap [p,q]$, where

$$B_\gamma := \{x|\, f(x) \leq \gamma - \varepsilon,\ g(x) \geq 0\}. \tag{7}$$

Since $g(x) = \min_{j=1,\ldots,m}\{u_j(x) - v_j(x)\}$ with $u_j(x), v_j(x)$ being increasing polynomials (see (2) and (3)), we can also write

$$B_\gamma = \{x|f(x) \leq \gamma - \varepsilon,\ u_j(x) - v_j(x) \geq 0\ \ j = 1,\ldots,m\}.$$

The reduction operation aims at replacing the box $[p,q]$ with a smaller box $[p',q'] \subset [p,q]$ without losing any point $x \in B_\gamma \cap [p,q]$, i.e. such that

$$B_\gamma \cap [p',q'] = B_\gamma \cap [p,q].$$

The box $[p', q']$ satisfying this condition is referred to as a *valid reduction* of $[p, q]$ and denoted by $\mathrm{red}[p, q]$.

In the following Lemma, e^i denotes the ith unit vector, i.e. a vector with 1 at the ith position and 0 everywhere else.

Lemma 1 (1) *If $f(p) > \gamma - \varepsilon$ or $\min_j\{u_j(q) - v_j(p)\} < 0$, then $B_\gamma \cap [p, q] = \emptyset$, i.e.* $\mathrm{red}[p, q] = \emptyset$.

(2) *If $f(p) \leq \gamma - \varepsilon$, and $\min_j\{u_j(q) - v_j(p)\} \geq 0$, then* $\mathrm{red}[p, q] = [p', q']$ *with*

$$p' = q - \sum_{i=1}^{n} \alpha_i (q_i - p_i) e^i, \qquad q' = p' + \sum_{i=1}^{n} \beta_i (q_i - p'_i) e^i, \tag{8}$$

where, for $i = 1, \ldots, n$,

$$\alpha_i = \sup\{\alpha |\, 0 < \alpha \leq 1,\ u_j(q - \alpha(q_i - p_i)e^i) \geq v_j(p),\ j = 1, \ldots, m\}, \tag{9}$$

$$\begin{aligned}\beta_i = \sup\{\beta |\, 0 < \beta \leq 1,\ &v_j(p' + \beta(q_i - p'_i)e^i) \leq u_j(q),\ j = 1, \ldots, m, \\ &f(p' + \beta(q_i - p'_i)e^i) \leq \gamma - \varepsilon\}.\end{aligned} \tag{10}$$

Proof (See e.g. [17] or [18])

Notice that, as can easily be verified, the box $[p', q'] = \mathrm{red}[p, q]$ still satisfies

$$f(p') \leq \gamma - \varepsilon, \quad \min_j\{u_j(q') - v_j(p')\} \geq 0. \tag{11}$$

5.2 Valid bounds

Given a box $M := [p, q]$, supposed to have been reduced, we want to compute an upper bound $\beta(M)$ for

$$\max\{g(x) |\, f(x) \leq \gamma - \varepsilon,\ x \in [p, q]\}. \tag{12}$$

Since $g(x) = \min_{j=1,\ldots,m}\{u_j(x) - v_j(x)\}$ and $u_j(x), v_j(x)$ are increasing, an obvious upper bound is

$$\min_{j=1,\ldots,m} [u_j(q) - v_j(p)]. \tag{13}$$

Although very simple, this bound suffices to ensure convergence of the algorithm, as will be shortly seen. However, for a better performance of the procedure, we can use any tight bound available. For instance, the following procedure may give a better bound.

The subproblem (12) can be rewritten as

$$\max\{x_{n+1} |\, x_{n+1} \leq u_k(x) - v_k(x)\ (k = 1, \ldots, m),\ f(x) \leq \gamma - \varepsilon,\ p \leq x \leq q\}.$$

Substituting (1) to $f(x)$ and (3) to $g_k(x) = u_k(x) - v_k(x)$, and introducing the additional variables $y_{ij} = x_i x_j, i \leq j$, we obtain the following linear program (LP)(M) in (x, x_{n+1}, y) as a linear relaxation of (12):

$$\begin{array}{lll} \multicolumn{3}{l}{\qquad \max_{x \in C \cap [p,q]}\ x_{n+1} \quad \text{s.t.}} \\ \sum_{i<j} c^k_{ij} y_{ij} + \sum_i c^k_i y_{ii} + \sum_i d^k_i x_i + b_k & \leq x_{n+1}, & k = 1, \ldots, m \\ \sum_{i<j} c_{ij} y_{ij} + \sum_i c_i y_{ii} + \sum_i d_i x_i & \leq \gamma - \varepsilon & \\ y_{ij} - p_j x_i - p_i x_j + p_i p_j & \geq 0, & \forall (i,j) \in K \\ y_{ij} - q_j x_i - q_i x_j + q_i q_j & \geq 0, & \forall (i,j) \in K \end{array}$$

$$\begin{array}{lll} y_{ij} - p_j x_i - q_i x_j + q_i p_j & \leq 0, & \forall (i,j) \in K \\ y_{ij} - q_j x_i - p_i x_j + p_i q_j & \leq 0, & \forall (i,j) \in K \\ y_{ii} - 2p_i x_i + p_i^2 & \geq 0, & \forall i \in N \\ y_{ii} - 2q_i x_i + q_i^2 & \geq 0, & \forall i \in N \\ y_{ii} - (p_i + q_i)x_i + p_i q_i & \leq 0, & \forall i \in N \\ y_{ii} - (p_i + q_i)x_i + (p_i + q_i)^2/4 & \geq 0, & \forall i \in N \\ p_i \leq x_i \leq q_i,\ i = 1, \ldots, n, & & \end{array}$$

where

$$K = \{(i,j) |\ 1 \leq i < j \leq n,\ c_{ij} \neq 0,\ \text{or } c_{ij}^k \neq 0 \text{ for some } k \in \{1, \ldots, m\}\}$$

$$N = \{i |\ 1 \leq i \leq n,\ c_i \neq 0,\ \text{or } c_i^k \neq 0 \text{ for some } k \in \{1, \ldots, m\}\}$$

An important property of LP(M) is that its optimal value $\beta(M)$ satisfies:

$$\max\{g(x) |\ f(x) \leq \gamma - \varepsilon,\ x \in [p,q]\} \leq \beta(M) \leq \min_{j=1,\ldots,m} [u_j(q) - v_j(p)]. \tag{14}$$

This follows from the fact that the constraints of LP(M) imply $p_i p_j \leq y_{ij} \leq q_i q_j\ \forall i,j$, and hence $\sum_{i<j} c_{ij}^k y_{ij} + \sum_i c_i^k y_{ii} + \sum_i d_i^k x_i + b_k \leq u_k(q) - v_k(p)\ \forall k = 1, \ldots, m$.

More generally, we shall show in the next section that any lower bound $\beta(M)$ satisfying (14) ensures convergence of the algorithm.

6 A robust algorithm

Incorporating the above BRB procedure for (Q/γ) into the SIT scheme yields the following robust algorithm for solving (Q):

SIT Algorithm for (Q)

Step 0 If no feasible solution is known, let $\gamma = f(b) + \varepsilon$; otherwise, let $\bar{x}$ be the best nonisolated feasible solution available, $\gamma = f(\bar{x})$. Let $\mathcal{P}_1 = \{M_1\}, M_1 = [a,b], \mathcal{R}_1 = \emptyset$. Set $k = 1$.

Step 1 For each box $M \in \mathcal{P}_k$:

(1) Compute its valid reduction red M.
(2) Delete M if red $M = \emptyset$.
(3) Replace M by red M if red $M \neq \emptyset$.
(4) If red$M = [p', q']$ then compute an upper bound $\beta(M)$ for $g(x)$ over the feasible solutions in red$M$. ($\beta(M)$ must satisfy $\beta(M) \leq \min_{j=1,\ldots,m}[u_j(q') - v_j(p')]$, see (14)). Delete M if $\beta(M) < 0$.

Step 1 Let $\mathcal{R}'_k = \mathcal{R}_k \cup \mathcal{P}'_k$.

Step 2 Let $\mathcal{P}'_k$ be the collection of boxes that results from $\mathcal{P}_k$ after completion of Step 1. Let $\mathcal{R}'_k = \mathcal{R}_k \cup \mathcal{P}'_k$.

Step 3 If $\mathcal{R}'_k = \emptyset$ then terminate: $\bar{x}$ is an ε-optimal solution of (Q) if $\gamma = f(\bar{x}) - \varepsilon$, or the problem (Q) is infeasible if $\gamma = f(b) + \varepsilon$.

Step 4 If $\mathcal{R}'_k \neq \emptyset$, let $[p^k, q^k] := M_k \in \operatorname{argmax}\{\beta(M) |\ M \in \mathcal{R}'_k\},\ \beta_k = \beta(M_k)$.

Step 5 If $\beta_k < \varepsilon$ then terminate: $\bar{x}$ is an essential ε-optimal solution of (Q) if $\gamma = f(\bar{x})$, or the problem (Q) is ε-essentially infeasible if $\gamma = f(b) + \varepsilon$.

Step 6 If $\beta_k \geq \varepsilon$, compute $\lambda_k = \max\{\alpha | f(p^k + \alpha(q^k - p^k)) \leq \gamma - \varepsilon\}$ and let

$$x^k = p^k + \lambda_k(q^k - p^k).$$

6a) If $g(x^k) > 0$ then x^k is a new nonisolated feasible solution of (Q) with $f(x^k) \leq \gamma - \varepsilon$: if $g(p^k) < 0$, compute the point $\bar{x}^k$ where the line segment joining p^k to x^k meets the surface $g(x) = 0$, and reset $\bar{x} \leftarrow \bar{x}^k$; otherwise, reset $\bar{x} \leftarrow p^k$. Go to Step 7.

6b) If $g(x^k) \leq 0$, go to Step 7, with $\bar{x}$ unchanged.

Step 7 Divide M_k into two subboxes by the standard bisection (or any bisection consistent with the bounding $M \mapsto \beta(M)$). Let $\mathcal{P}_{k+1}$ be the collection of these two subboxes of M_k, $\mathcal{R}_{k+1} = \mathcal{R}'_k \setminus \{M_k\}$. Increment k, and return to Step 1.

Proposition 2 *The above algorithm terminates after finitely many steps, yielding either an essential ε-optimal solution of* (*Q*)*, or an evidence that the problem is essentially infeasible.*

Proof Since any feasible solution x with $f(x) \leq \gamma - \varepsilon = f(\bar{x}) - \varepsilon$ must lie in some box $M \in \mathcal{R}'_k$ the event $\mathcal{R}'_k = \emptyset$ implies that no such solution exists, hence the conclusion in Step 3. If Step 5 occurs, so that $\beta_k < \varepsilon$, then $\max(Q/\gamma) < \varepsilon$, hence the conclusion in Step 5 (see Proposition 1). Thus the conclusions in Steps 3 and 5 are correct. It remains to show that either Steps 3 ($\mathcal{R}'_k = \emptyset$) or 5 ($\beta_k < \varepsilon$) must occur for sufficiently large k. To this end, observe that in Step 6, since $f(p^k) \leq \gamma - \varepsilon$ (see (11)), the point x^k exists and satisfies $f(x^k) \leq \gamma - \varepsilon$, so if $g(x^k) > 0$, then by Proposition 1, x^k is a nonisolated feasible solution with $f(x^k) \leq f(\bar{x}) - \varepsilon$, justifying Step 6a (note that $\bar{x}^k$ is a nonisolated feasible solution at least as good as x^k). Suppose now that the algorithm is infinite. Since each occurrence of Step 6a decreases the current best value at least by $\varepsilon > 0$ while $f(x)$ is bounded below it follows that Step 6a cannot occur infinitely often. Consequently, for all k sufficiently large, $\bar{x}$ is unchanged, and $g(x^k) \leq 0$, while $\beta_k \geq \varepsilon$. But, as $k \to +\infty$, we have, by exhaustiveness of the subdivision, $\mathrm{diam} M_k \to 0$, i.e. $\|q^k - p^k\| \to 0$. Denote by $\tilde{x}$ the common limit of q^k and p^k as $k \to +\infty$. Since

$$\varepsilon \leq \beta_k \leq \min_{j=1,\ldots,m} [u_j(q^k) - v_j(p^k)]$$

it follows that

$$\varepsilon \leq \lim_{k \to +\infty} \beta_k \leq \min_{j=1,\ldots,m} [u_j(\tilde{x}) - v_j(\tilde{x})] = g(\tilde{x}).$$

But by continuity, $g(\tilde{x}) = \lim_{k \to +\infty} g(x^k) \leq 0$, a contradiction. Therefore, the algorithm must be finite. □

Remark 1 The SIT Algorithm works its way to the optimum through a sequence of better and better nonisolated solutions. If for some reason the algorithm has to be stopped prematurely, some reasonably good feasible solution may have been already obtained. This is one of its advantages over most existing algorithms, which may be useless when stopped prematurely.

Remark 2 For regular problems (i.e. problems with no isolated feasible solutions), an essential optimal solution is obviously an optimal solution in the usual sense. From the computational complexity point of view the present SIT algorithm does not differ much from the monotonic branch-reduce-and-bound algorithm developed in [17].

Case where there are equality constraints
So far we assumed (4), so that the feasible set has a nonempty interior. We now extend the method to the case when there are equality constraints, e.g.

$$h_l(x) = 0, \quad l = 1, \ldots, s \tag{15}$$

(so that assumption (4) fails).

First, if some of these constraints are linear, they can be used to eliminate certain variables. Therefore, without loss of generality we can assume that all the constraints (15) are *nonlinear*. Since, however, in the most general case one cannot expect to compute a solution of a nonlinear system of equations in finitely many steps, one should be content with an approximate system

$$-\delta \leq h_l(x) \leq \delta, \quad l = 1, \ldots, s,$$

where $\delta > 0$ is the tolerance. In other words, a set of constraints of the form

$$\begin{aligned} g_j(x) &\geq 0, \quad j = 1, \ldots, m, \\ h_l(x) &= 0, \quad l = 1, \ldots, s, \end{aligned}$$

should be replaced by the approximate system

$$\begin{aligned} g_j(x) &\geq 0, \quad j = 1, \ldots, m, \\ h_l(x) + \delta &\geq 0, \quad l = 1, \ldots, s, \\ -h_l(x) + \delta &\geq 0, \quad l = 1, \ldots, s. \end{aligned}$$

The method presented in the previous sections can then be applied to the resulting approximate problem. With $g(x) = \min_{k=1,\ldots,m} g_k(x)$, $h(x) = \max_{l=1,\ldots,s} |h_l(x)|$, the required assumption is, instead of (4), $\{x \in [a,b] | \ g(x) > 0, -h(x) + \delta > 0\} \neq \emptyset$. An essential ε-optimal solution to the above defined approximate problem is then called an *essential* (δ, ε)*-optimal solution* of (P).

Remark 3 Since any boolean constraint of the form $x \in \{0,1\}^n$ can be rewritten as a system of quadratic constraints:

$$x_i(x_i - 1) \geq 0, \quad 0 \leq x_i \leq 1, \ \ i = 1, \ldots, n$$

any boolean optimization problem can in principle be transformed into a quadratic optimization problem. Nevertheless, such a problem should not be studied as a (QQP) because its feasible set consists only of isolated points, among which the optimal solution should be selected. For these problems a special approach based on monotonic optimization is currently available. (see [18])

7 Illustrative example and numerical results

The SIT algorithm, coded in C++, with CPLEX 8.0 for solving linear programs, was tested on a number of nontrivial instances of problem (QQP) taken from the literature. Computational experiments were made on a PC Pentium IV 2.53GHz, RAM 256Mb DDR.

1. An interesting and instructive example is furnished by the following problem (to be referred to as problem (A)) studied in [2]:

$$\begin{aligned}
\min_x \; & z(x) := (12.626260(x_{12}+x_{13}+x_{14}+x_{15}+x_{16}) \\
& \qquad\qquad -1.231059(x_1x_{12}+x_2x_{13}+x_3x_{14}+x_4x_{15}+x_5x_{16})) \\
\text{s.t. } & 50 \le z(x) \le 250; \\
& -3.475x_i + 100x_j + .0975x_i^2 - 9.75x_ix_j \le 0 \quad i=1,2,3,4,5, \; j=i+5; \quad (*) \\
& -x_6x_{11} + x_7x_{11} - x_1x_{12} + x_6x_{12} \ge 0, \qquad (**) \\
& 50x_7 - 50x_8 - x_1x_{12} + x_2x_{13} + x_7x_{12} - x_8x_{13} = 0, \qquad (***) \\
& 50x_8 + 50x_9 - x_2x_{13} + x_3x_{14} + x_8x_{13} - x_9x_{14} \le 500, \\
& -50x_9 + 50x_{10} - x_3x_{14} + x_4x_{15} - x_8x_{15} + x_9x_{14} \le 0, \\
& 50x_4 - 50x_{10} - x_4x_{15} - x_4x_{16} + x_5x_{16} + x_{10}x_{15} \le 0,
\end{aligned}$$

$$\begin{array}{lll}
50x_4 - x_4x_{16} + x_5x_{16} \ge 450, & & -x_1 + 2x_7 \le 1, \\
x_1 \le x_2 \le x_3 \le x_4 \le x_5, & x_6 \le x_7, & x_8 \le x_9 \le x_{10} \le x_4, \\
0 \le x_{11} - x_{12} \le 50, & 0.001 \le x_6 \le 1, & \\
1 \le x_1 \le 8.03773157, & 1 \le x_7 \le 4.51886579, & 10^{-7} \le x_{12} \le 100, \\
1 \le x_2 \le 9, & 1 \le x_8 \le 9, & 1 \le x_{13} \le 50, \\
4.5 \le x_3 \le 9, & 1 \le x_9 \le 9, & 50 \le x_{14} \le 100, \\
4.5 \le x_4 \le 9, & 1 \le x_{10} \le 9, & 50 \le x_{15} \le 100, \\
9 \le x_5 \le 10, & 0.1 \le x_{11} \le 100, & 10^{-7} \le x_{16} \le 50.
\end{array}$$

Following the RLT method, by introducing the additional variables $y = (y_{ij})$, with

$$y_{ij} = x_ix_j \quad (i,j = 1,\ldots,n, \; i \le j) \tag{16}$$

a (linear program) LP can be derived from problem (A) augmented with some implied constraints, by replacing each polynomial

$$f_k(x) = \sum_{i<j} c_{ij}^k x_ix_j + \sum_i c_i^k x_i^2 + \sum_i d_i^k x_i$$

with its linearization

$$g_k(x,y) = \sum_{i<j} c_{ij}^k y_{ij} + \sum_i c_i^k y_{ii} + \sum_i d_i^k x_i.$$

Problem (A) is thus equivalent to the LP with the additional nonconvex constraints (16). A feasible solution (x,y) to LP satisfying

$$|y_{ij} - x_ix_j| \le \varepsilon_r \quad (i,j = 1,\ldots,n, i \le j) \tag{17}$$

is called an ε_r-approximate solution of (A). In other words, an ε_r-approximate solution is a feasible solution of the problem $\mathrm{LP}_{\varepsilon_r}$ obtained by appending the constraints (17) to LP . An ε_r-approximate solution (x^*, y^*) is called an $(\varepsilon_r, \varepsilon_z)$-optimal solution if it is an ε_z-optimal solution of the problem $\mathrm{LP}_{\varepsilon_r}$.

By using a well devised branch and cut algorithm to solve problem (A), the following $(\varepsilon_r, \varepsilon_z)$-optimal solution, for $\varepsilon_r = \varepsilon_z = 10^{-5}$, with objective function value 174.788, has been found in [2]:

$$\begin{aligned}
x^* = (&8.03773, 8.161, 9, 9, 9, 1, 1.07026, 1.90837, \\
&1.90837, 1.90837, 50.5042, .504236, 7.26387, 50, 50, 0).
\end{aligned}$$

However, this solution turned out to be infeasible, since it violates the constraints (*) for $i = 1, 3, 4, 5$, and the constraint (**), with an error far exceeding the tolerance.

Furthermore, the $(\varepsilon_r, \varepsilon_z)$-optimal value 174.788 found in [2] is very far from the true optimum. In fact, solving problem (A) by the SIT algorithm with tolerances $\delta = 10^{-6}, \varepsilon = 10^{-2}$, yields an (δ, ε)-optimal solution right at the first iteration

$$\hat{x} = (1, 9, 9, 9, 10, 0.001, 1, 1.156863, 1.156863, 1.156863, 0.1, 0, 1, 50, 50, 0)$$

with objective function value 156.219629 which is much inferior to 174.788.

All this once more illustrates the weakness of outer approximation methods for nonconvex global optimization and the inadequacy of the concept of ε-approximate solution as defined in [2].

Note that problem (A) is derived from a test problem in [3] by several modifications. These modifications change an originally convex problem (a geometric program) into a nonconvex one. Specifically one inequality constraint is changed into equality (***), while five inequality constraints are reversed. By restoring the original inequality constraints, but keeping the equality (***), the problem is still nonconvex because of the presence of this equality. Solving this nonconvex quadratic problem by the SIT algorithm, with tolerances $\delta = 10^{-6}, \varepsilon = 10^{-5}$, yields, after 418 iterations, an essential (δ, ε)-optimal solution

$$\begin{aligned}\bar{x} = (&8.036214, 8.153175, 9, 9, 9, 0.999183, 1.065565, 1.908367,\\ &1.908367, 1.908367, 49.854883, 0.470290, 7.272981, 50, 50, 0)\end{aligned}$$

with objective function value 174.789648. These results agree with those reported in [3] for the original geometric program—which is to be expected. In fact, since the inequality constraint in the geometric program that has become an equality constraint in the nonconvex quadratic problem is in fact satisfied as equality by the optimal solution of the geometric program, this change does not actually affect the optimal solution. Computation by the SIT algorithm required 51.953 sec. and went through 165 cycles of incumbent transcending.

2. Aside from the above problem from Audet et al. [2], we also tested the SIT algorithm on a number of problems taken from Floudas et al. [5]. The results of these preliminary experiments are reported in the table below. Columns n and m indicate the number of variables and the number of constraints, respectively. Column Iteration indicates the number of iterations, Cycle: number of cycles of incumbent transcending, and Time: running time (in seconds). Tolerances: $\delta = 10^{-}6, \varepsilon = 0.01$.

Prob.	n	m	Iteration	Cycle	Time	Source
1	5	5	33	27	0.156	[5] 7.2.6
2	5	6	117	21	0.641	[5] 7.2.5
3	8	6	47	16	0.672	[5] 3.2
4	8	7	2,874	422	39.625	[5] 5.2.4
5	9	7	10	6	0.219	[5] 5.2.2, case 1
6	9	7	42	31	0.859	[5] 5.2.2, case 2
7	9	7	10	4	0.234	[5] 5.2.2, case 3
8	9	12	115	33	2.313	[5] 7.2.2
9	17	22	110	12	12.39	[2]
10	22	25	47	8	12.61	[5] 5.3.2

References

1. Al-Khayyal, F.A., Larsen, C., van Voorhis, T.: A relaxation method for nonconvex quadraticaly constrained quadratic programs. J. Glob. Optim. **6**, 215–230 (1995)
2. Audet, C., Hansen, P., Jaumard, B., Savard, G.: A branch and cut algorithm for nonconvex quadratically constrained quadratic programming. Math. Program. Ser. A **87**, 131–152 (2000)
3. Dembo, R.S.: A set of geometric programming test problems and their solutions. Math. Program. **10**, 192–213 (1976)
4. Floudas, C.A.: Deterministic Global Optimization. Kluwer, Dordrecht (1999)
5. Floudas, C.A. et al.: Handbook of Test Problems in Local and Global Optimization. Kluwer, Dordrecht (1999)
6. Horst, R., Tuy, H.: Global Optimization: Deterministic Approaches, 3rd edn. Springer, Berlin (1996)
7. Lasserre, J.: Global optimization with polynomials and the problem of moments. SIAM J. Optim. **11**, 796–817 (2001)
8. Phong, T.Q., Tao, P.D., Hoai An, L.T.: A method for solving d.c. programming problems. application to fuel mixture nonconvex optimization problem. J. Glob. Optim. **6**, 87–105 (1995)
9. Sherali, H.D., Tuncbilek, C.H.: A global opimization algorithm for polynomial programming using a reformulation-linearization technique. J. Glob. Optim. **2**, 101–112 (1992)
10. Sherali, H.D., Tuncbilek, C.H.: A reformulation-convexification approach for solving nonconvex quadratic programming problems. J. Glob. Optim. **7**, 1–31 (1995)
11. Shor, N.Z., Stetsenko, S.I.: Quadratic Extremal Problems and Nondifferentiable Optimization. Naukova Dumka, Kiev in Russian (1989)
12. Shor, N.Z.: Nondifferentiable Optimization and Polynomial Problems. Kluwer, Dordrecht (1998)
13. Tuy, H.: Convex Analysis and Global Optimization. Kluwer, Dordrecht (1998)
14. Tuy, H.: Monotonic optimization: problems and solution approaches. SIAM J. Optim. **11**(2), 464–494 (2000)
15. Tuy, H., Al-Khayyal, F., Thach, P.T.: Monotonic optimization: branch and cut methods. In: Audet, C., Hansen, P., Savard, G. (eds.) Essays and Surveys on Global Optimization, pp. 39–78. Springer, Berlin (2005)
16. Tuy, H.: Robust solution of nonconvex global optimization problems. J. Glob. Optim. **32**, 307–323 (2005)
17. Tuy, H.: Polynomial optimization: a robust approach. Pac. J. Optim. **1**, 357–374 (2005)
18. Tuy, H., Minoux, M., Hoai-Phuong, N.T.: Discrete monotonic optimization with application to a discrete location problem. SIAM J. Optim. **17**, 78–97 (2006)